BuildTech Overview

Research and Market Opportunities as a natural biproduct of the success of Cargo

# Overview

We’ve missed the concept in Build and Package systems. They are more closely to Software Development, Product, and Distribution Cycle Management Platforms is the more correct way to look at it and, in that, a hope might emerge to arrive at a place of correctness. Language Agnosticism is an implied requirement in this. Platform Agnosticism is as well.

## Product vs Operations

We’ve learned that, Operations wins when its viewed as and treated as a product. The Hello Fresh Open Source Platform in nearly 90% of its total signature as an open-source deliverable in order to force a certain kind of architecture and process-pipeline workflow internally allowing HF to scale into any market with 10’s of thousands or sourcing partner-providers on their weekly food delivery packages is a strong illustration of this difference. Had it been manual binding into the information systems of new providers for new markets, they would have never been able to scale with the rate that their marketing and ad-tech allowed them to scale the business and this would have meant significant competition present in foreign markets prior to them introducing themselves there.

# Build Systems Analysis

GNU Platform Build: Make, AutoTools, etc

CMake

Ninja

Maven

Bazel

# Package Management Systems Analysis

## Systems Package Managers

RPM

Brew

Port

https://en.wikipedia.org/wiki/List\_of\_build\_automation\_software

* [Apache Ant](https://en.wikipedia.org/wiki/Apache_Ant), popular for [Java](https://en.wikipedia.org/wiki/Java_(software_platform)) platform development and uses an [XML](https://en.wikipedia.org/wiki/XML) file format
* Apache Buildr, historic open-source build system, [Rake](https://en.wikipedia.org/wiki/Rake_(software))-based, gives the full power of scripting in [Ruby](https://en.wikipedia.org/wiki/Ruby_(programming_language)) with integral support for most abilities wanted in a build system
* [Apache Maven](https://en.wikipedia.org/wiki/Apache_Maven), a Java platform tool for dependency management and automated software build
* [ASDF](https://en.wikipedia.org/wiki/Another_System_Definition_Facility) LISP build system for building LISP projects
* [A-A-P](https://en.wikipedia.org/wiki/A-A-P), a [Python](https://en.wikipedia.org/wiki/Python_(programming_language))-based build tool
* [Bazel](https://en.wikipedia.org/wiki/Bazel_(software)), a portion of Blaze ([Google's](https://en.wikipedia.org/wiki/Google) own build tool) written in Java, using Starlark (BUILD file syntax) to build projects in Java, C, C++, Go, Python, Objective-C, and others
* [BitBake](https://en.wikipedia.org/wiki/BitBake), a Python-based tool with the special focus of distributions and packages for embedded Linux cross-compilation
* [Boot](https://en.wikipedia.org/wiki/Boot_(software)), a [Java](https://en.wikipedia.org/wiki/Java_(software_platform)) build and dependency management tool written in [Clojure](https://en.wikipedia.org/wiki/Clojure)
* [boost.build](https://en.wikipedia.org/wiki/Boost_(C%2B%2B_libraries)) For C++ projects, cross-platform, based on [Perforce Jam](https://en.wikipedia.org/wiki/Perforce_Jam)
* [Buck](https://en.wikipedia.org/wiki/Buck_(software)), a build system developed and used by Facebook, written in Java, using Starlark (BUILD file syntax) as [Bazel](https://en.wikipedia.org/wiki/Bazel_(software)" \o "Bazel (software))
* [Buildout](https://en.wikipedia.org/wiki/Buildout), a Python-based build system for creating, assembling and deploying applications from multiple parts
* [Cabal](https://en.wikipedia.org/wiki/Cabal_(software)), a common architecture for building applications and libraries in the programming language [Haskell](https://en.wikipedia.org/wiki/Haskell_(programming_language))
* [Dub](https://en.wikipedia.org/wiki/D_(programming_language)#Development_tools), the official package and build manager of the [D](https://en.wikipedia.org/wiki/D_(programming_language)) Language
* dale,[[2]](https://en.wikipedia.org/wiki/List_of_build_automation_software#cite_note-2) a D build tool
* FinalBuilder, for Windows software developers. FinalBuilder provides a graphical IDE to both create and run build projects in a single application. The final builder also includes the ability the execute the unit test, deploy web projects or install and test applications.
* [Flowtracer](https://en.wikipedia.org/wiki/Flowtracer), a build management tool
* [Gradle](https://en.wikipedia.org/wiki/Gradle), an open-source build and automation system with an [Apache Groovy](https://en.wikipedia.org/wiki/Groovy_(programming_language))-based [domain specific language](https://en.wikipedia.org/wiki/Domain_specific_language) (DSL), combining features of [Apache Ant](https://en.wikipedia.org/wiki/Apache_Ant) and [Apache Maven](https://en.wikipedia.org/wiki/Apache_Maven) with additional features like a reliable incremental build
* [Grunt](https://en.wikipedia.org/wiki/Grunt_(software)), a build tool for front-end web development
* [Gulp](https://en.wikipedia.org/wiki/Gulp.js), a build tool for front-end web development
* [IncrediBuild](https://en.wikipedia.org/wiki/IncrediBuild), a suite of grid computing software for compiling and building software
* [Leiningen](https://en.wikipedia.org/wiki/Leiningen_(software)), a tool providing commonly performed tasks in Clojure projects, including build automation
* [Mix](https://en.wikipedia.org/wiki/Mix_(build_tool)), the [Elixir](https://en.wikipedia.org/wiki/Elixir_(programming_language)) build tool
* [MSBuild](https://en.wikipedia.org/wiki/MSBuild), the [Microsoft](https://en.wikipedia.org/wiki/Microsoft) build engine
* [NAnt](https://en.wikipedia.org/wiki/NAnt), a tool similar to Ant for the [.NET Framework](https://en.wikipedia.org/wiki/.NET_Framework)
* [Ninja](https://en.wikipedia.org/wiki/Ninja_(build_system)), a small build system focused on speed by using build scripts generated by higher-level build systems
* [Perforce Jam](https://en.wikipedia.org/wiki/Perforce_Jam), a build tool by [Perforce](https://en.wikipedia.org/wiki/Perforce), inspired by Make
* [Qt Build System](https://en.wikipedia.org/wiki/Qbs_(build_tool))
* [Rake](https://en.wikipedia.org/wiki/Rake_(software)), a [Ruby](https://en.wikipedia.org/wiki/Ruby_(programming_language))-based build tool
* rez,[[3]](https://en.wikipedia.org/wiki/List_of_build_automation_software#cite_note-3) a C/C++ build tool
* [sbt](https://en.wikipedia.org/wiki/Sbt_(software)), a build tool built on a [Scala](https://en.wikipedia.org/wiki/Scala_(programming_language))-based DSL
* [SCons](https://en.wikipedia.org/wiki/SCons), Python-based, with integrated functionality similar to [autoconf](https://en.wikipedia.org/wiki/Autoconf" \o "Autoconf)/[automake](https://en.wikipedia.org/wiki/Automake" \o "Automake)
* [Stack](https://en.wikipedia.org/wiki/Stack_(Haskell)), a tool to build [Haskell](https://en.wikipedia.org/wiki/Haskell_(programming_language)) projects, manage their dependencies (compilers and libraries), and for testing and benchmarking.
* tinyrick,[[4]](https://en.wikipedia.org/wiki/List_of_build_automation_software#cite_note-4) a Rust build tool
* vast,[[5]](https://en.wikipedia.org/wiki/List_of_build_automation_software#cite_note-5) a shell build tool
* [Visual Build](https://en.wikipedia.org/wiki/Visual_Build), a [graphical user interface](https://en.wikipedia.org/wiki/Graphical_user_interface) software for software builds
* [Waf](https://en.wikipedia.org/wiki/Waf_(build_system)), a Python-based tool for configuring, compiling and installing applications. It is a replacement for other tools such as [Autotools](https://en.wikipedia.org/wiki/Autotools" \o "Autotools), [Scons](https://en.wikipedia.org/wiki/Scons" \o "Scons), [CMake](https://en.wikipedia.org/wiki/CMake" \o "CMake) or Ant
* BuildAMation, a multi-platform tool, using a declarative syntax in C# scripts, that builds C/C++ code in a terminal using multiple threads, or generates project files for Microsoft Visual Studio, Xcode or MakeFiles.
* [CMake](https://en.wikipedia.org/wiki/CMake) generates files for various build tools, such as [make](https://en.wikipedia.org/wiki/Make_(software)), [ninja](https://en.wikipedia.org/wiki/Ninja_(build_system)), Apple's [Xcode](https://en.wikipedia.org/wiki/Xcode" \o "Xcode), and [Microsoft Visual Studio](https://en.wikipedia.org/wiki/Microsoft_Visual_Studio).[[6]](https://en.wikipedia.org/wiki/List_of_build_automation_software#cite_note-6) CMake is also directly used by some [IDE](https://en.wikipedia.org/wiki/Integrated_development_environment) as [Qt Creator](https://en.wikipedia.org/wiki/Qt_Creator),[[7]](https://en.wikipedia.org/wiki/List_of_build_automation_software#cite_note-7) [KDevelop](https://en.wikipedia.org/wiki/KDevelop" \o "KDevelop) and [GNOME Builder](https://en.wikipedia.org/wiki/GNOME_Builder).[[8]](https://en.wikipedia.org/wiki/List_of_build_automation_software#cite_note-phoronix.com-8)
* [GNU Build System](https://en.wikipedia.org/wiki/GNU_Build_System) (aka autotools), a collection of tools for portable builds. These in particular include [Autoconf](https://en.wikipedia.org/wiki/Autoconf" \o "Autoconf) and [Automake](https://en.wikipedia.org/wiki/Automake" \o "Automake), cross-unix-platform tools that together generate appropriate localized makefiles.
* [GYP](https://en.wikipedia.org/wiki/GYP_(software)) (Generate Your Projects) - Created for [Chromium](https://en.wikipedia.org/wiki/Chromium_(web_browser)); it is another tool that generates files for the native build environment. It has been superseded by GN which generates files for [ninja](https://en.wikipedia.org/wiki/Ninja_(build_system)) and other build systems.
* [imake](https://en.wikipedia.org/wiki/Imake)
* [Meson](https://en.wikipedia.org/wiki/Meson_(software)), a build system optimized for performance and usability is based on [ninja](https://en.wikipedia.org/wiki/Ninja_(build_system)) on Linux, [Visual Studio](https://en.wikipedia.org/wiki/Visual_Studio) on Windows and [Xcode](https://en.wikipedia.org/wiki/Xcode" \o "Xcode) on macOS. Meson is also directly used by [GNOME Builder](https://en.wikipedia.org/wiki/GNOME_Builder).[[8]](https://en.wikipedia.org/wiki/List_of_build_automation_software#cite_note-phoronix.com-8)
* [OpenMake Software](https://en.wikipedia.org/wiki/OpenMake_Software) Meister
* [Premake](https://en.wikipedia.org/wiki/Premake), a Lua-based tool for making makefiles, Visual Studio files, Xcode projects, and more
* [qmake](https://en.wikipedia.org/wiki/Qmake)
* [AnthillPro](https://en.wikipedia.org/wiki/AnthillPro), build automation with pipeline support for deployment automation and testing. Cross-platform, cross-language
* [Apache Continuum](https://en.wikipedia.org/wiki/Apache_Continuum) - discontinued
* [Bamboo](https://en.wikipedia.org/wiki/Bamboo_(software)), continuous-integration software
* [Bitbucket](https://en.wikipedia.org/wiki/Bitbucket) Pipelines and Deployments, continuous integration for [Bitbucket](https://en.wikipedia.org/wiki/Bitbucket) hosted repositories[[9]](https://en.wikipedia.org/wiki/List_of_build_automation_software#cite_note-9)
* [Buildbot](https://en.wikipedia.org/wiki/Buildbot), a Python-based software development continuous-integration tool which automates the compile/test cycle
* [CruiseControl](https://en.wikipedia.org/wiki/CruiseControl), for Java and .NET
* [Go continuous delivery](https://en.wikipedia.org/wiki/Go_continuous_delivery), open source, cross-platform
* [GitLab](https://en.wikipedia.org/wiki/GitLab) ([GitLab Runner](https://en.wikipedia.org/wiki/GitLab)), continuous integration and [git](https://en.wikipedia.org/wiki/Git) server
* [GitHub](https://en.wikipedia.org/wiki/GitHub) ([GitHub Actions](https://en.wikipedia.org/wiki/GitHub)), free continuous integration service for open-source projects and [git](https://en.wikipedia.org/wiki/Git) server
* [Hudson](https://en.wikipedia.org/wiki/Hudson_(software)), an extensible continuous-integration engine
* [Jenkins](https://en.wikipedia.org/wiki/Jenkins_(software)), an extensible continuous-integration engine, [forked](https://en.wikipedia.org/wiki/Fork_(software_development)) from Hudson
* [Spinnaker](https://en.wikipedia.org/wiki/Spinnaker_(software)), open source multi-cloud continuous delivery service from Netflix and Google
* [TeamCity](https://en.wikipedia.org/wiki/TeamCity)
* [Travis CI](https://en.wikipedia.org/wiki/Travis_CI), a hosted continuous-integration service
* [Ansible](https://en.wikipedia.org/wiki/Ansible_(software)) (Python-based)
* [CFEngine](https://en.wikipedia.org/wiki/CFEngine)
* [Chef](https://en.wikipedia.org/wiki/Chef_(software)) (Ruby-based)
* [LCFG](https://en.wikipedia.org/wiki/LCFG)
* [NixOS](https://en.wikipedia.org/wiki/NixOS) Declarative configuration model
* [OpenMake Software](https://en.wikipedia.org/wiki/OpenMake_Software) Release Engineer
* [Otter](https://en.wikipedia.org/wiki/Otter_(software))
* [Puppet](https://en.wikipedia.org/wiki/Puppet_(software)) (Ruby-based)
* [Salt](https://en.wikipedia.org/wiki/Salt_(software)) (Python-based)
* [Rex](https://en.wikipedia.org/wiki/Rex_(software)) (Perl-based)

A meta-build tool is capable of building many different projects using a subset of existing build tools. Since these usually provide a list of packages to build, they are also often called [package managers](https://en.wikipedia.org/wiki/Package_manager).

* [Pkgsrc](https://en.wikipedia.org/wiki/Pkgsrc), [Portage](https://en.wikipedia.org/wiki/Portage_(software)), [MacPorts](https://en.wikipedia.org/wiki/MacPorts" \o "MacPorts) and other package managers derived from the [BSD Ports Collection](https://en.wikipedia.org/wiki/Ports_collection).
* [Nix](https://en.wikipedia.org/wiki/Nix_package_manager), functional package manager for Linux and macOS focusing on [reproducible builds](https://en.wikipedia.org/wiki/Reproducible_build), used for the [NixOS](https://en.wikipedia.org/wiki/NixOS" \o "NixOS) Linux distribution.
* [Guix](https://en.wikipedia.org/wiki/GNU_Guix), functional package manager based on Nix, used for the [GuixSD](https://en.wikipedia.org/wiki/GuixSD" \o "GuixSD) Linux distribution.
* [Collective Knowledge](https://en.wikipedia.org/wiki/Collective_Knowledge_(software)), cross-platform package manager to rebuild software environment for research workflows
* [Homebrew](https://en.wikipedia.org/wiki/Homebrew_(package_management_software)), package manager for macOS

| **Tool name** | **Description language** | **License** |
| --- | --- | --- |
| [**A-A-P**](https://en.wikipedia.org/wiki/A-A-P) | recipe | [GNU GPL](https://en.wikipedia.org/wiki/GNU_General_Public_License) |
| [**Ant**](https://en.wikipedia.org/wiki/Apache_Ant) | [XML](https://en.wikipedia.org/wiki/XML) | [Apache License 2.0](https://en.wikipedia.org/wiki/Apache_License_2.0) |
| [**AnthillPro**](https://en.wikipedia.org/wiki/AnthillPro) | Wraps Make, Ant, Maven, MsBuild, Nant, etc. for controlled build, deploy, test processes. | Discontinued |
| [**Bamboo**](https://en.wikipedia.org/wiki/Atlassian_bamboo) | continuous integration | [Trialware](https://en.wikipedia.org/wiki/Trialware) |
| [**Bazel**](https://en.wikipedia.org/wiki/Bazel_(software)) | BUILD/Starlark, a Python-like DSL | [Apache License 2.0](https://en.wikipedia.org/wiki/Apache_License_2.0) |
| **BuildAMation** | [C#](https://en.wikipedia.org/wiki/C_Sharp_(programming_language)) for build scripts, [XML](https://en.wikipedia.org/wiki/XML) for high level dependencies | [New BSD License](https://en.wikipedia.org/wiki/New_BSD_License) |
| **Buildr** | [Ruby](https://en.wikipedia.org/wiki/Ruby_(programming_language)) | [Apache License 2.0](https://en.wikipedia.org/wiki/Apache_License_2.0) |
| [**Boot**](https://en.wikipedia.org/wiki/Boot_(software)) | Clojure | [Eclipse Public License](https://en.wikipedia.org/wiki/Eclipse_Public_License) |
| [**Capistrano**](https://en.wikipedia.org/wiki/Capistrano_(software)) | [XML](https://en.wikipedia.org/wiki/XML) | [MIT License](https://en.wikipedia.org/wiki/MIT_License) |
| [**CMake**](https://en.wikipedia.org/wiki/CMake) | uses CMakeLists.txt file | [New BSD License](https://en.wikipedia.org/wiki/New_BSD_License) |
| [**Collective Knowledge Framework**](https://en.wikipedia.org/wiki/Collective_Knowledge_(software)) | [Python](https://en.wikipedia.org/wiki/Python_(language)) scripts with [JSON](https://en.wikipedia.org/wiki/JSON) API and [JSON](https://en.wikipedia.org/wiki/JSON) meta-description | [New BSD License](https://en.wikipedia.org/wiki/New_BSD_License) |
| [**Continuum**](https://en.wikipedia.org/wiki/Apache_Continuum) | ? | [Apache License 2.0](https://en.wikipedia.org/wiki/Apache_License_2.0) |
| [**CruiseControl**](https://en.wikipedia.org/wiki/CruiseControl) | [XML](https://en.wikipedia.org/wiki/XML) | [BSD](https://en.wikipedia.org/wiki/BSD_licenses)-style license |
| [**Dub**](https://en.wikipedia.org/wiki/D_(programming_language)#Development_tools) | [JSON](https://en.wikipedia.org/wiki/JSON), [SDL](https://en.wikipedia.org/wiki/Specification_and_Description_Language) | [MIT License](https://en.wikipedia.org/wiki/MIT_License) |
| **FinalBuilder** | graphical IDE with support for Ant/NAnt, MSBuild, JScript, VBScript, [IronPython](https://en.wikipedia.org/wiki/IronPython" \o "IronPython), [PowerShell](https://en.wikipedia.org/wiki/Windows_PowerShell) | [Trialware](https://en.wikipedia.org/wiki/Trialware) |
| [**Gradle**](https://en.wikipedia.org/wiki/Gradle) | Groovy-based DSL; [Kotlin](https://en.wikipedia.org/wiki/Kotlin_(programming_language))-based DSL | [Apache License 2.0](https://en.wikipedia.org/wiki/Apache_License_2.0) |
| [**Jenkins**](https://en.wikipedia.org/wiki/Jenkins_(software)) | continuous integration | [MIT License](https://en.wikipedia.org/wiki/MIT_License) |
| [**Homebrew**](https://en.wikipedia.org/wiki/Homebrew_(package_management_software)) | [Ruby](https://en.wikipedia.org/wiki/Ruby_(programming_language)) | [Simplified BSD License](https://en.wikipedia.org/wiki/Simplified_BSD_License) |
| [**Leiningen**](https://en.wikipedia.org/wiki/Leiningen_(software)) | Clojure | [Eclipse Public License](https://en.wikipedia.org/wiki/Eclipse_Public_License) |
| [**make**](https://en.wikipedia.org/wiki/Make_(software)) | uses Makefile | Same as the bundling OS |
| [**Maven**](https://en.wikipedia.org/wiki/Apache_Maven) | [Project Object Model](https://en.wikipedia.org/wiki/Project_Object_Model) | [Apache License 2.0](https://en.wikipedia.org/wiki/Apache_License_2.0) |
| [**Meson build system**](https://en.wikipedia.org/wiki/Meson_build_system) | custom DSL | [Apache License 2.0](https://en.wikipedia.org/wiki/Apache_License_2.0) |
| [**MPW Make**](https://en.wikipedia.org/wiki/Macintosh_Programmer%27s_Workshop#Other_tools) | ? | [Freeware](https://en.wikipedia.org/wiki/Freeware) |
| [**MSBuild**](https://en.wikipedia.org/wiki/MSBuild) | [XML](https://en.wikipedia.org/wiki/XML) | [MIT License](https://en.wikipedia.org/wiki/MIT_License) |
| [**NAnt**](https://en.wikipedia.org/wiki/NAnt) | [XML](https://en.wikipedia.org/wiki/XML) | [GNU GPL](https://en.wikipedia.org/wiki/GNU_General_Public_License) |
| [**nmake**](https://en.wikipedia.org/wiki/Nmake) | uses Makefile | [Freeware](https://en.wikipedia.org/wiki/Freeware) |
| [**Open Build Service**](https://en.wikipedia.org/wiki/Open_Build_Service) | uses various package and image description formats (spec, dsc, ARCH, kiwi) | [GNU GPL](https://en.wikipedia.org/wiki/GNU_General_Public_License) |
| [**Perforce Jam**](https://en.wikipedia.org/wiki/Perforce_Jam) | uses Jamfile | Discontinued |
| [**Rake**](https://en.wikipedia.org/wiki/Rake_(software)) | Ruby | [MIT License](https://en.wikipedia.org/wiki/MIT_License) |
| [**sbt (Simple Build Tool)**](https://en.wikipedia.org/wiki/SBT_(software)) | Scala-based DSL | [New BSD License](https://en.wikipedia.org/wiki/New_BSD_License) |
| [**SCons**](https://en.wikipedia.org/wiki/SCons) | [Python](https://en.wikipedia.org/wiki/Python_(programming_language)) | [MIT License](https://en.wikipedia.org/wiki/MIT_License) |
| [**Team Foundation Server**](https://en.wikipedia.org/wiki/Team_Foundation_Server) | MSBuild, Windows Presentation Foundation, JSON - REST interfaces, Programmatic (Can generate definitions through code) | [Trialware](https://en.wikipedia.org/wiki/Trialware) |
| [**Visual Build**](https://en.wikipedia.org/wiki/Visual_Build) | [XML](https://en.wikipedia.org/wiki/XML) | [Trialware](https://en.wikipedia.org/wiki/Trialware) |
| [**Waf**](https://en.wikipedia.org/wiki/Waf_(build_system)) | [Python](https://en.wikipedia.org/wiki/Python_(programming_language)) | [New BSD License](https://en.wikipedia.org/wiki/New_BSD_License) |

# Dependency Problem Space Analysis